Bulletin of Electrical Engineering and Informatics
Vol. 12, No. 6, December 2023, pp. 3880∼3888
ISSN: 2302-9285, DOI: 10.11591/eei.v12i6.6110

On-off attack detection in trust model using intra-daily variability for the IoT

Sornalakshmi Kannan[1], Revathi Venkataraman[2], Gowri Sankar Ramachandran[3]
[1]Department of Computing Technologies, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Tamilnadu, India
[2]School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Tamilnadu, India
[3]School of Computer Science, Queensland University of Technology, Brisbane, Australia

Article Info

Article history:

Received Feb 28, 2023
Revised May 3, 2023
Accepted Jun 4, 2023

Keywords:

Internet of things
Intra-daily variability
On-off attack
Trust computation

ABSTRACT

The growth of the internet of things (IoT) increases the need to develop the trust computational model for heterogeneous networks with various IoT devices. Trust models are considered as an effective tool to mitigate insider attacks induced by IoT devices. However, trust models are exposed to on-off attacks, in which devices randomly exhibit good and bad behaviors to avoid being categorized as low-trust devices. The objective of this work is to recognize the malicious devices executing on-off attacks in IoT applications. This paper introduces an on-off attack detection strategy for the trust computational model based on the non-parametric index named intra-daily variability (IV). IV indicates trust fragmentation which depends on the frequency and the transitions between periods of low and high trust values of a device. The higher value of IV indicates the occurrence of fragmented trust values and the lower value of IV indicates the occurrence of non-fragmented trust values. Experimental results show that the proposed model outperforms the baseline methods by increasing the on-off attack detection rate.

Corresponding Author:

Sornalakshmi Kannan
Department of Computing Technologies, School of Computing, College of Engineering and Technology
SRM Institute of Science and Technology
Kattankulathur 603203, Tamilnadu, India
Email: sornak@srmist.edu.in

1. INTRODUCTION

The internet of things (IoT) helps to establish heterogeneous environments where smart devices with different processing and storage capabilities can cooperate and communicate to provide services to the user [1]-[3]. The trust computational models play an essential role in IoT since it determines whether a specific IoT device can provide a satisfactory service to the user and can identify the malicious IoT devices presented in the network [4]-[6]. Since the trust management system detects misbehaving devices effectively, it becomes an attractive target for adversaries. Attacks such as conflicting behavior attacks, on-off attacks, sybil attacks, and bad-mouthing attacks can undermine the accuracy of the trust management systems [7]-[10]. In an on-off attack, a malicious device exhibits legitimate and malicious behavior randomly to avoid being categorized as a low-trusted device [11]-[13]. During the on state, the device will exhibit a malicious attack; while during the off state, the device will exhibit normal behavior. Various methods such as counting the number of attacks performed within a particular time frame, dynamic sliding window strategy, and machine learning methods are used to detect the on-off attacks. Each method has its merits and demerits; the related works section discusses these.

In this paper, we present an intra-daily variability (IV) based method to detect the nodes exhibiting on-off attacks in the trust management system. In chronobiology, IV is used to measure circadian rhythm disturbance, where circadian rhythm is a 24-hour pattern that regulates the physical, mental, and behavioral changes. One typical example of circadian rhythm is sleeping during at night and awake during the day. IV is used to measure rhythm fragmentation [14], [15], i.e. taking naps at the day and awake at night. This IV concept has been used to detect the fragmentation of the trust values of a node. It provides the value that defines how much the on-off attack affects the trust model. IV is calculated using the current and previous trust values of the node. In our work, since IV calculation is done in the cloud, more previous trust values can be considered for calculating IV. This IV method gives higher attack detection rate than existing methods in detecting the on-off attack. The remaining section of the paper is arranged as follows: the background of the proposed work is described in section 2, in section 3 discussed the works related to on-off attack detection, section 4 presents the proposed method of detecting on-off attacks in trust evaluation systems, the performance of the proposed method is evaluated and the results are presented in section 5, and the conclusion is drawn in section 6.

2. BACKGROUND

2.1. On-off attack

The on-off attack cycle has been discussed in [16], [17]. This attack cycle comprises two periods, namely on and off. During the ”on” period the device launches an attack, hence the trust value of the device would be low. Similarly during the ”off” period the device behaves normal, hence the trust value of the device would be high. Duration of the ”on” and ”off” periods can be of different or equal length based on the device’s attack strategy. Figure 1 depicts the on-off attack cycle.

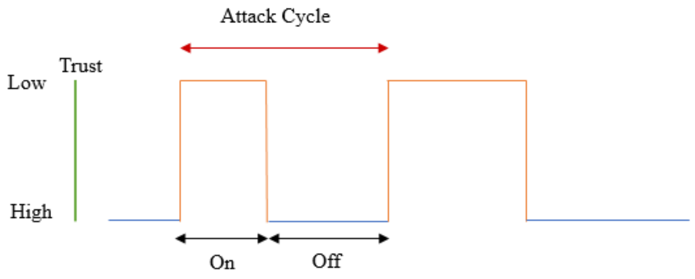

Figure 1. On-off attack

2.2. Trust management system

The devices in the network are assumed to be static and each device has a unique identifier. Also, the devices in the network are assumed to have enough energy and communication power to transmit the data. It is assumed that the trust model can distinguish temporary errors and malicious behavior. Trust computational models compute the trustworthiness of the devices participating in the application scenario. For example, smart healthcare applications such as circadian rhythm monitoring may contain smart devices such as smart beds, smart lights, and smart doors. While using these devices to monitor health and to make automated decisions based on the data received from these devices, the trustworthiness of the devices that send data plays an important role. Hence, it is necessary to ensure that the trustworthiness of the device is high. A secure and not compromised device would constantly produce reliable data. Whereas, a compromised device would produce inconsistent data exhibiting anomaly [18]. It is the responsibility of the trust computational model to detect the devices that exhibit malicious behavior by reducing their trust value. In cases where malicious devices perform an on-off attack, it is possible for the trust model to classify the malicious device as legitimate. To avoid such misclassification, the trust values computed by the trust model for a particular device must be analyzed to find the presence of an on-off attack. The block diagram of a trust management system is presented in Figure 2. The basic components of a trust management system include:

- Evidence collector and evidence database: it collects the evidence needed to verify the trustworthiness of the device and verifies it with the stored evidence. The result is passed to the trust calculation engine.
- Trust calculation engine: it computes the trust value of each smart device based on verified evidence.
- Trust store: it stores the trust values of each device computed at different time intervals. Since our trust management system resides in the cloud, it could store past trust values of a device to a great extent.
- On-off attack detector: it is responsible to detect the on-off attack behavior of each device. It calculates the IV using the previously stored trust values and the current trust value. Based on the IV value, it evaluates the risk factor of the on-off attack behavior of a node. The appropriate decision is delivered as an output from this component. Based on this, the device may be evicted if the IV value is high.

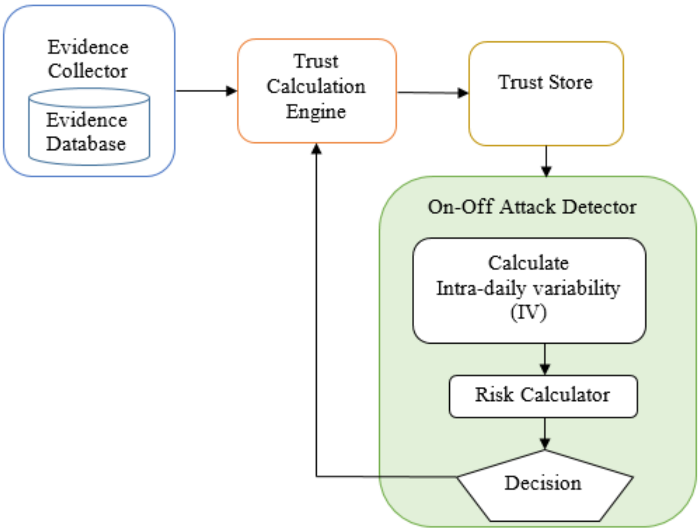

Figure 2. Trust management system

3. RELATED WORKS

Many research works have addressed the on-off attack detection strategy. Suryani *et al.* [19] proposed an algorithm to detect the on-off attacks. The statistical methods are used to identify the IoT object exhibiting attack behavior. In this work, the on-off attack is considered as an accumulation of trust-based attacks such as good-mouthing and bad-mouthing attacks. The identity of the object and the number of attacks performed in a certain period are recorded. If the total number of trust-related attacks exceeds three, the IoT device would be identified as malicious and performing an on-off attack.

Chae *et al.* [20] proposed a defending strategy to detect the neighboring nodes which perform on-off attacks using predictability trust. This predictability trust uses the node's current trust to predict future behavior and checks the consistency of its current behavior with the past behavior. If the node's behavior is inconsistent, it is suspected as on-off attack. It uses two types of sliding windows such as static and dynamic for normal and malicious behaviors. The bad behavior window (BBW) can detect six good behaviors (6G) and one bad behavior (1B) on-off attack, i.e., 86%G and 14%B on-off attack. Since the length of BBW could not meet the requirement for detecting 9G and 1B attack, the on-off attack would be permitted when the off-to-on ratio exceeds a certain level.

Sahoo *et al.* [21] presented a method to detect the on-off attack based on sliding window and misbehavior history. The node x monitors the behavior of node y and keeps track of the number of successful $(S_{x,y})$ and unsuccessful interactions $(U_{x,y})$ in each time unit (t_i). To provide the defense for the trust management system from the on-off attack, the direct trust is computed using $S_{x,y}$, $U_{x,y}$, and the misbehavior frequency of the node. The frequency of misbehavior (freq Δt) denotes how frequently a node exhibits malicious behavior within a time frame.

Caminha *et al.* [22] proposed the smart trust management system using machine learning to detect on-off attackers. An elastic slide window is used to discriminate the broken and misbehaving nodes. Farea and

Küçük [23] introduced the model which uses machine learning to identify on-off attacks in the IoT with the help of artificial neural networks (ANN). ANN multilayer perceptron (MLP) is used to find any inconsistent change by learning from the recurring characteristics of each node. Based on the features of the node, the data is classified into either a normal or malicious node. According to Labraoui *et al.* [24], a trust model named "O^2 trust" to detect the on-off attack is proposed. It maintains the misbehavior history of each node participating in the network. It uses the penalty policy against this misbehavior history. A technique called predictability trust to detect the malicious nodes performing the on-off attacks is proposed in [25]. It uses the sliding window concept to count the number of normal and malicious behaviors of each node. Predictability trust measures the number of normal behaviors of a node out of total behaviors. The node is identified as malicious if the predictability trust is not greater than the threshold. Similarly, a trust models for wireless sensor networks that can mitigate on-off attacks is proposed in [16], [24]. The misbehavior history and the aggregated misbehavior component are used for trust estimation, which helps to detect on-off attacks.

The previous work considers the number of attacks launched within a particular time. If it exceeds the threshold, the node is identified as performing an on-off attack. Also, since it uses the sliding window concept to track the misbehavior of the node, the definition of the sliding window becomes a challenging and critical requirement. In addition, the existing machine-learning methods classify the nodes as malicious or normal. Unlike existing methods, this work uses a statistical method "IV" to detect the fragmented behavior of the trust values, that is the rate of transition between the high and low. The current research involves designing a robust trust management system that detects the on-off attack.

4. PROPOSED METHOD

This section describes our proposed method IV to detect on-off attacks. Circadian rhythms are physical, mental and behavioral changes that follow a 24-hour cycle [26], [27]. Circadian rhythm is comprised of three different variables: i) interdaily stability (IS)-it shows the strength of the circadian rhythm by measuring the degree of consistency of the activity patterns; ii) IV-it shows the circadian rhythm disturbance by measuring the frequency and the transitions between rest and activity; and iii) amplitude (AMP)-the difference between least active 5-hour (L5) and most active 10-hour (M10) patterns in an average 24-hour period [28]. In these three variables, studies have shown that IV is an excellent variable to analyze, as it is used as a marker of sleep-wake disruption. If there is a prolonged period of high activity or a continuous period of low or no activity within 24 hours, the lowest value of IV occurs. If there is a greater degree of fragmentation in the rest-activity pattern, a higher IV value occurs [29]. The formula for IV is shown in (1):

$$IV = \frac{n\sum_{i=2}^{n}(x_i - x_{i-1})^2}{(n-1)\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{1}$$

Where n represents the total number of data points, $\bar{x}$ is the mean of all the data points, and x_i represents the individual data point. It is calculated as the ratio of the sum of squared differences between consecutive trust values to the sum of squared differences between each trust value and the mean trust values.

Inspired by IV in the circadian rhythm context, we estimate the anomalous behavior of devices in the network and use that to evict the compromised devices quickly. The formula to detect IV has been used to detect the on-off attack in the trust model. In our case, the trust value of the device computed at different time intervals has been used to compute the IV. IV indicates trust fragmentation which depends on the frequency and the transitions between periods of low and high trust values of a device on a test basis. The higher value of IV indicates the occurrence of fragmented trust values and the lower value of IV indicates the occurrence of non-fragmented trust values. The trust values computed by the trust computational model range from 0 to 1. The trust values greater than the threshold are mapped to 1. i.e., high and the trust values less than the threshold are mapped to 0. i.e., low. The values 1 and 0 are mapped to high and low trust, respectively. When there is no fragmentation, i.e. if all trust values are zero (malicious) or all trust values are one (legitimate), the IV value reaches zero. It denotes that there is no fragmented behavior of a device. i.e. the device is either malicious or legitimate, but it has not shown on-off attack behavior. When there are fragmented trust values, i.e. if trust values are zero and one alternatively, IV reaches high values, it denotes that there is fragmented behavior of a device, i.e. the device is exhibiting both legitimate and malicious behavior. The on-off attack detection and IV computation method are presented in Algorithm 1 and Algorithm 2, respectively.

Algorithm 1 IV based on-off attack detection

```
Input: Trust values
Output: Decision about on-off attack detection
PROCEDURE:
for each decision epoch do
    for each Device $x$ belongs to Device Set $D$ do
        Compute Intra-daily Variability $IV_t^x$ by Algorithm 2
        If $IV_t^x > T_{th}$
            Decision: The device $x$ is identified as malicious since it exhibits an On-Off attack
        Else
            Decision: The device $x$ is not exhibiting an On-Off attack
return Decision about On-Off attack
```

Algorithm 2 IV computation

```
Input: Trust values-$T$
Output: IV value
PROCEDURE:
for $T$ do
    mean.counts = mean($T$)
    numerator = sum$(diff(T)^2)/(length(T) - 1)$
    denominator = sum$((T - mean.counts)^2)/length(T)$
    IV = numerator/denominator
return IV
```

The IV calculation and how it relates to trust fragmentation is explained using example trust values listed in Table 1. We have considered 12 trust values to calculate IV. Hence, the value of n is 12. In the case 1-trust values, $\bar{x}$ is the mean of all the data. Hence, $\bar{x} = 0.5$. Since the trust values are more fragmented, i.e., the transitions between low and high trust values, the IV reaches the highest value of 4.

$$IV = \frac{\begin{array}{r}12 \times ((1-0)^2 + (0-1)^2 + (1-0)^2 + (0-1)^2 + (1-0)^2 + (0-1)^2 + \\ (1-0)^2) + (0-1)^2 + (1-0)^2 + (0-1)^2 + (1-0)^2)\end{array}}{\begin{array}{r}11 \times ((0-0.5)^2 + (1-0.5)^2 + (0-0.5)^2 + (1-0.5)^2 + (0-0.5)^2 + (1-0.5)^2 \\ + (0-0.5)^2 + (1-0.5)^2 + (0-0.5)^2 + (1-0.5)^2 + (0-0.5)^2 + (1-0.5)^2)\end{array}}$$

$$IV = \frac{132}{33}$$

$$IV = 4$$

In case 2-trust values, since the trust values are not fragmented, i.e., there are no transitions between low and high trust values, the IV reaches the lowest value of 0. In case 3-trust values, since the trust values are moderately fragmented, i.e., there are some transitions between low and high trust values, the IV reaches the value of 2.18. Similarly, in case 4-trust values, since the trust values are less fragmented, i.e., high off-to-on ratio, the IV reaches the value of 0.65. If the IV value exceeds zero, it is considered an on-off attack.

Table 1. Sample data for IV calculation

Timestamp (x_i)	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}
Case 1-trust values	0	1	0	1	0	1	0	1	0	1	0	1
Case 2-trust values	1	1	1	1	1	1	1	1	1	1	1	1
Case 3-trust values	1	1	0	0	0	0	1	1	0	1	0	1
Case 4-trust values	0	0	0	0	0	0	0	0	0	0	1	1

4.1. Attack detection time

The length of the data, i.e, the total number of trust values computed during different time intervals, passed as input to the IV is an important factor in detecting on-off attacks. Since IV detection is done in cloud, the storage, and processing time would not be a challenge. Hence, all the IV values computed must be considered to take decision about node eviction. If all the IV values are high, the node could be evicted, or the trust management system could use the computed IV value to compute the new trust value. This will make the trust management system as an adaptive system.

4.2. Intra-daily variability threshold

In the trust context, the IV value ranges from zero to four, where zero indicates trust values are not fragmented and four indicates trust values are fragmented. In the circadian rhythm context, the standard value of IV in the elderly must be less than 1.10. It does not mean that the IV value less than 1.10 exhibit no fragmentation; instead, it implies less fragmentation and will not affect the circadian rhythm much. Similarly, the standard value, i.e. IV threshold, for on-off attack detection must be defined to meet the security requirement. In our proposed method, IV value that equals zero is only considered non-fragmented behavior. If it exceeds zero, it is considered as an on-off attack. Though static threshold can be accessed after many trials and testing, dynamic threshold and dynamic time duration for IV calculation will reduce further attack possibilities. The reason is that when static threshold and static time duration is used, it would be easy for an attacker to predict the behavior of our on-off attack detection model. To avoid this, dynamic threshold and time duration can be defined in the on-off attack detection model.

5. RESULT AND DISCUSSION

The performance of the proposed approach was tested using real-time experimentation. The real-time experimentation setup was developed with the Raspberry Pi4, which emerged as an IoT platform installed with the Raspbian OS. The sensors, such as environment temperature, light, sound, and passive infrared (PIR) are integrated into the Raspberry Pi. The data from these sensors are sent to the server. To test the on-off attack detection using IV, a flooding attack was initiated on the IoT device using Python, and the traffic data was collected using Wireshark. The traffic data available in pcap file are exported into csv format. From the traffic data, many features, such as no. of packet-in, no. of packet-out, in packet length, and attack status were retrieved. Only attack status is used to find the trust label from these many features. The attack status contains 0 or 1, i.e. 0 indicates normal behavior and 1 indicates attack behavior. In addition to this attack status, we included a trust label in this dataset. The trust label contains 0 or 1, i.e. 0 indicates malicious behavior and 1 indicates good/normal behavior. This real-time data collected around 32,400 seconds (9 hours) was used as input to the ”ActCR” R programming library, which has an inbuilt IV function to find the on-off attack. The trust labels computed at different time intervals are fed as input to this model. When trust values oscillate between one and zero, the IV values reach high. It represents that the trust values are fragmented, leading to the detection of an on-off attack. The experimental parameters are shown in Table 2.

Table 2. Experimental parameters

Parameters	Values
IoT device	Raspberry Pi4
No. of devices	2
Duration	9 hours
Attack	Flooding attack

Figure 3 shows the changes in IV values for different time frames. The legend represents the time frame used to calculate the IV values. The legend ”last 1 h” shows IV values calculated using data from the last 1 hour, with 0 indicating no on-off attack and 1 indicating an on-off attack. Similarly, the legend ”last 2 h”, ”last 3 h”, and ”last 4 h” shows IV values calculated using data from the last 2 hours, last 3 hours, and last 4 hours, respectively. The graph shows that the IV values increase when there is an on-off attack (more fragmented behavior of a device) and decrease when there is no or less on-off attack (no or less fragmented behavior of a device). By comparing these, we can see how the IV values change over time and how they are affected by the time frame used for the calculation.

From Figure 4, we infer that the proposed method IV provides 90% and 80% attack detection rate for IV threshold values 0.6 and 0.7 respectively. As the baseline methods proposed in [24] fail to detect few on-off attacks, it gives 58% and 67% attack detection rate for threshold values 0.6 and 0.7 respectively. Similarly, the other baseline method proposed in [16] gives 40% and 62% attack detection rate for threshold values 0.6 and

0.7 respectively. The reason is the false negative rate, i.e., the number of on-off attacks detected as non-on-off attacks or normal behavior. In the baseline methods, if the trust threshold value increases, the on-off attack detection rate also increases. However in our proposed method, if the IV threshold decreases, the on-off attack detection rate increases and if the IV threshold increases, the on-off attack detection rate decreases. In the baseline method, the values greater than the trust threshold are considered trustworthy (off state) and those less than the trust threshold are considered untrustworthy (on state). Our proposed method uses all the trust values computed by the trust model to find the value of transitions between high and low trust values. A higher IV value indicates higher fragmentation in the trust values and implying an on-off attack. Conversely, a lower IV value indicates no fragmentation in the trust values. The reason for the increased attack detection rate in our proposed method is the use of IV to detect the on-off attack.

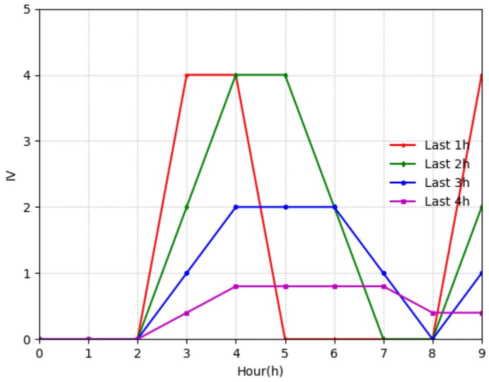

Figure 3. Last hours vs IV

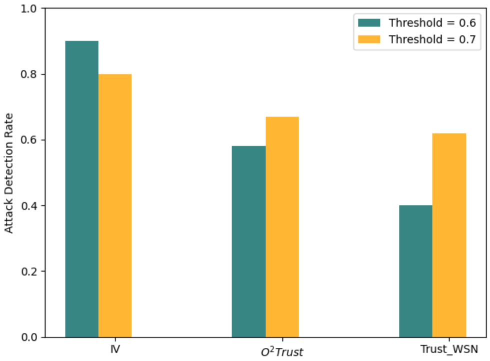

Figure 4. Attack detection rate

According to Nasution *et al.* [30], if the number of good-mouthing and bad-mouthing attacks exceeds three, the device is marked as malicious for performing an on-off attack., i.e. if three transactions performed by the device exhibit three good-mouthing attacks or three bad-mouthing attacks, it will also be recognized as an on-off attack. In our proposed model, if three transactions performed by the device exhibit three good-mouthing attacks or three bad-mouthing attacks, it will be recognized as non-fragmented behavior., i.e. the device is not exhibiting two different behaviors. Hence, the IV value would be zero. It states that the device continuously exhibits good-mouthing or bad-mouthing attack behavior.

Similarly, suppose two transactions performed by the device exhibit two good-mouthing attacks or two bad-mouthing attacks out of three transactions. In that case, it will not be recognized as the on-off attack in the method proposed in [30]. Since it exhibits fragmented behavior (i.e. trust labels - 011, 101, or 110), our proposed model effectively identifies this as an on-off attack. For example, for the case-4 trust values, the method proposed in [30] could not identify the on-off attack.

6. CONCLUSION

On-off attacks threaten the security of IoT trust models through devices performing good and bad random behavior. IV helps to detect this fragmented behavior of the devices. The proposed method has been applied to the data collected from an IoT device to detect the on-off attack. The results show that the proposed method outperforms the base methods in attack detection rate. The results inferred from this statistical analysis can be incorporated into the trust management systems to provide better protection against malicious attacks.

ACKNOWLEDGEMENT

The authors would like to thank Hamid Alhamadi, associate professor of Computer Science, Kuwait University for his valuable suggestions during this work.

REFERENCES

[1] C. V. L. Mendoza and J. H. Kleinschmidt, “Mitigating on-off attacks in the internet of things using a distributed trust management scheme,” *International Journal of Distributed Sensor Networks*, vol. 11, no. 11, pp. 1–8, 2015, doi: 10.1155/2015/859731.

[2] G. Paolone, D. Iachetti, R. Paesani, F. Pilotti, M. Marinelli, and P. D. Felice, “A Holistic Overview of the Internet of Things Ecosystem,” *Internet of Things*, vol. 3, no. 4, pp. 398–434, 2022, doi: 10.3390/iot3040022.

[3] A. Heidari and M. A. J. Jamali, “Internet of Things intrusion detection systems: a comprehensive review and future directions,” *Cluster Computing*, pp. 1–28, 2022, doi: 10.1007/s10586-022-03776-z.

[4] C. Lewis, N. Li, and V. Varadharajan, “Targeted Context-Based Attacks on Trust Management Systems in IoT,” *IEEE Internet of Things Journal*, vol. 10, no. 14, pp. 12186–12203, 2023, doi: 10.1109/JIOT.2023.3245605.

[5] M. P. Lokhande and D. D. Patil, “Trust Computation Model for IoT Devices Using Machine Learning Techniques,” in *Proceeding of First Doctoral Symposium on Natural Computing Research*, 2021, pp. 195–205, doi: 10.1007/978-981-33-4073-2_20.

[6] I. -R. Chen, J. Guo, D. -C. Wang, J. J. P. Tsai, H. A. -Hamadi, and I. You, “Trust as a Service for IoT Service Management in Smart Cities,” in *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, 2018, pp. 1358–1365, doi: 10.1109/HPCC/SmartCity/DSS.2018.00225.

[7] Y. Sun, Z. Han, W. Yu, and K. R. Liu, “Attacks on Trust Evaluation in Distributed Networks,” in *2006 40th Annual Conference on Information Sciences and Systems*, 2006, pp. 1461–1466, doi: 10.1109/CISS.2006.286695.

[8] B. Pourghebleh, K. Wakil, and N. J. Navimipour, “A Comprehensive Study on the Trust Management Techniques in the Internet of Things,” *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9326–9337, 2019, doi: 10.1109/JIOT.2019.2933518.

[9] A. Sharma, E. S. Pilli, A. P. Mazumdar, and P. Gera, “Towards trustworthy Internet of Things: A survey on Trust Management applications and schemes,” *Computer Communications*, vol. 160, pp. 475–493, 2020, doi: 10.1016/j.comcom.2020.06.030.

[10] H. A. -Hamadi, I. R. Chen, and J. H. Cho, “Trust Management of Smart Service Communities,” *IEEE Access*, vol. 7, pp. 26362–26378, 2019, doi: 10.1109/ACCESS.2019.2901023.

[11] W. Fang, M. Xu, C. Zhu, W. Han, W. Zhang, and J. J. P. C. Rodrigues, “FETMS: Fast and Efficient Trust Management Scheme for Information-Centric Networking in Internet of Things,” *IEEE Access*, vol. 7, pp. 13476–13485, 2019, doi: 10.1109/ACCESS.2019.2892712.

[12] F. Moradi, A. Sedaghatbaf, S. A. Asadollah, A. Causevic, and M. Sirjani, “On-Off Attack on a Blockchain-based IoT System,” in *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2019, pp. 1768–1773, doi: 10.1109/ETFA.2019.8868238.

[13] D. He, C. Chen, S. Chan, J. Bu, and A. V. Vasilakos, “ReTrust: Attack-resistant and lightweight trust management for medical sensor networks,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 4, pp. 623–632, 2012, doi: 10.1109/TITB.2012.2194788.

[14] D. B. Constantino, N. B. Xavier, R. Levandovski, T. Roenneberg, M. P. Hidalgo, and L. K. Pilz, “Relationship Between Circadian Strain, Light Exposure, and Body Mass Index in Rural and Urban Quilombola Communities,” *Frontiers in Physiology*, vol. 12, pp. 1–11, 2022, doi: 10.3389/fphys.2021.773969.

[15] K. Sornalakshmi, R. Venkataraman, N. Shalin, M. J. Samrai, and M. Viveka, “Rhythm Monitor-A Wearable for Circadian Health Monitoring,” in *2022 International Conference on Electronic Systems and Intelligent Computing (ICESIC)*, 2022, pp. 341–344, doi: 10.1109/ICESIC53714.2022.9783609.

[16] F. Ishmanov, S. W. Kim, and S. Y. Nam, “A secure trust establishment scheme for wireless sensor networks,” *Sensors*, vol. 14, no. 1, pp. 1877–1897, 2014, doi: 10.3390/s140101877.

[17] Y. Chae, L. C. Dipippo, and Y. L. Sun, “Predictability trust for Wireless Sensor Networks to provide a defense against On/off attack,” in *8th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom)*, 2012, pp. 406–415.

[18] F. Azzedin and M. Ghaleb, “Internet-of-things and information fusion: Trust perspective survey,” *Sensors*, vol. 19, no. 8, pp. 1–22, 2019, doi: 10.3390/s19081929.

[19] V. Suryani, S. Sulistyo, and Widyawan, “The Detection of On-Off Attacks for the Internet of Things Objects,” in *2018 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, 2018, pp. 1–5, doi: 10.1109/ICCEREC.2018.8712098.

[20] Y. Chae, L. C. DiPippo, and Y. L. Sun, “Trust management for defending on-off attacks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 4, pp. 1178–1191, 2015, doi: 10.1109/TPDS.2014.2317719.

[21] R. R. Sahoo, S. Sarkar, and S. Ray, “Defense Against On-Off Attack in Trust Establishment Scheme for Wireless Sensor Network,” in *2019 2nd International Conference on Signal Processing and Communication (ICSPC)*, 2019, pp. 153–160, doi: 10.1109/ICSPC46172.2019.8976869.

[22] J. Caminha, A. Perkusich, and M. Perkusich, “A Smart Trust Management Method to Detect On-Off Attacks in the Internet of Things,” *Security and Communication Networks*, vol. 2018, pp. 1–10, 2018, doi: 10.1155/2018/6063456.

[23] A. H. Farea and K. Küçük, “Enhancement Trust Management in IoT to Detect ON-OFF Attacks with Cooja,” *International Journal of Multidisciplinary Studies and Innovative Technologies*, vol. 5, no. 2, pp. 123–128, 2021.

[24] N. Labraoui, M. Gueroui, and L. Sekhri, “On-off attacks mitigation against trust systems in wireless sensor networks,” in *Computer Science and Its Applications*, Cham: Springer, 2015, pp. 406–415, doi: 10.1007/978-3-319-19578-0_33.

[25] S. M. Sony and S. B. Sasi, “On-off attack management based on trust,” in *2016 Online International Conference on Green Engineering and Technologies (IC-GET)*, 2016, pp. 1–4, doi: 10.1109/GET.2016.7916760.

[26] “Circadian Rhythms,” National Institute of General Medical Sciences, 2020. [Online]. Available: https://nigms.nih.gov/education/fact-sheets/Pages/Circadian-Rhythms.aspx. Access date: 04 May 2022

[27] K. Datta, “Physiology of Circadian Rhythm,” in *Making Sense of Sleep Medicine*, Florida, US: CRC Press, 2022, pp. 9–13, doi: 10.1201/9781003093381-3.

[28] E. J. W. Van Someren, A. Kessler, M. Mirmiran, and D. F. Swaab, “Indirect bright light improves circadian rest-activity rhythm disturbances in demented patients,” *Biological Psychiatry*, vol. 41, no. 9, pp. 955–963, 1997, doi: 10.1016/S0006-3223(97)89928-3.

[29] B. S. B. Gonçalves, P. R. A. Cavalcanti, G. R. Tavares, T. F. Campos, and J. F. Araujo, "Nonparametric methods in actigraphy: An update," *Sleep Science*, vol. 7, no. 3, pp. 158–164, 2014, doi: 10.1016/j.slsci.2014.09.013.

[30] A. P. Nasution, V. Suryani, and A. A. Wardana, "IoT Object Security towards On-off Attack Using Trustworthiness Management," in *2020 8th International Conference on Information and Communication Technology (ICoICT)*, 2020, pp. 1–6, doi: 10.1109/ICoICT49345.2020.9166169.

BIOGRAPHIES OF AUTHORS

Sornalakshmi Kannan received the B.Tech. degree in information technology from SCAD College of Engineering and Technology, Tirunelveli District, India, in 2011, the M.Tech. degree in Computer Science and Engineering from the SRM Institute of Science and Technology (SRMIST/formerly known as SRM University), Kattankulathur, Chennai, in 2017. She is currently pursuing her Ph.D. degree in computer science and engineering in SRMIST. Also she is currently working as a research associate with the Department of Computing Technologies, SRMIST, Chennai. Her current research interests include wireless sensor networks, internet of things, security, and trust management. She can be contacted at email: sornak@srmist.edu.in.

Revathi Venkataraman is currently a professor and chairperson in the School of Computing SRMIST, India. She received her Ph.D. degree from the SRMIST. Her research interests include trust computing, cyber security, security enhancements, and privacy considerations for IoT. She has received funding from Defence Research and Development Organization. She has also patented a few of her innovative ideas in wireless networking. She can be contacted at email: revathin@srmist.edu.in.

Gowri Sankar Ramachandran is a research fellow at the Trusted Networks Lab, Queensland University of Technology. He has completed his Ph.D. from KU Leuven, Belgium. After his Ph.D., he worked as a postdoctoral researcher and senior research associate at the University of Southern California, USA, between 2017 and 2020. His research interests revolve around the blockchain, supply chains, and IoT. He is broadly interested in solving privacy and trust issues in decentralised architectures. He has served on the organisation committees of ACM and IEEE conferences such as ICBC, SenSys, IoTDI, IPSN, EWSN, and BlockSys. He can be contacted at email: g.ramachandran@qut.edu.au.